REMARKS

Claims 1-42 were examined and rejected in this case. Claims 31-37 have been deleted. Claims 43-50 have been added. Claims 1-30 and 38-50 are currently pending. Reconsideration of the application is respectfully requested.

Applicant requests that the Examiner enter the above amendments to the Specification. No new matter is being added.

Applicant further requests that the Examiner enter amendments to Figures 2, 3A and 6. Accordingly, Applicant has attached copies of Figures 2, 3A and 6 with the changes marked in red and a letter to the chief draftsman.

Before addressing the substantive objections and rejections, a brief review of Applicant's invention is helpful. Applicant's invention is directed at a system and method for maintaining a central site, i.e., the master server, which stores service-enabling connection, communication and configuration data. Accordingly, a user at a remote site can contact the central site, and can request a connection to a particular service. The central site downloads a corresponding Downloadable to enable the remote site to establish a connection to and communicate with the service. The system and method enable various modes of accessing the service, namely, through a direct connection to the service or through the server acting as a proxy to the service. The user can then control the service from the remote site. The central site downloads configuration information to the remote site, so that the remote site can be configured to look and feel in a user-preferred way. The system and method enable connection to a service from a remote client without carting any software, without carting service addresses, and without carting user-specific configuration data. The user need only locate any remote client which has a Downloadable code execution engine, such as an applet engine.

35 USC § 112 Objection to the Specification

Generally speaking, the Examiner appears to be focusing on implementation details such as servlets and applets. Applicant is not trying to claim applets or servlets! Applicant is claiming various systems and methods for enabling a roaming user to access services from any kiosk in any country just by sitting at a terminal having a Downloadable code execution engine such as an applet engine. The "how" to make these systems and methods of the present invention, e.g., the protocols needed to communicate in a network, the Downloadable code executable in a distributed environment (applets, servlets, CGI programs, plugins), and the locations of files for configuring the remote client (user interface, bookmarks available, e-mail addresses), is known. The "what" to make is new.

In paragraph 2, the Examiner objected to the Specification under 37 CFR § 1.71 as failing the written description and enablement requirements. The Examiner asserted that Applicant did not teach details of the communications engine and the applet engine as recited in claims 31-34, 36-38, 40 and 41, details of the configuration engine as recited in claim 25, or details of the applet host engine as recited in claim 42.

On page 11, lines 14-16, the Specification describes a communications engine as an engine that generates and transfers message packets to and from the Internet via a communications interface. Claim 3 as filed indicates that establishing a communications link between a client and a server includes opening an Internet protocol connection. Applicant submits that one skilled in the art readily knew different protocols, including circuit switching and packet switching, for communicating information across a network. Similarly, one skilled in the art readily knew the conventional widely used TCP/IP internet protocol, and that this protocol includes rules for format, error handling, handshaking, switch failures, etc. No undue experimentation to implement a communications engine would have been necessary.

On page 12, lines 5-6, the Specification describes an applet engine that handles and executes downloaded applets. In the Specification, Applicant cited Netscape Navigator, Internet Explorer and the Java Developers Kit as examples of browsers having an applet engine. Applet-enabled web browsers were available from at least as early as 1995, i.e., from about two years before this application was filed. For Java environments, these 1995 browsers included a Java Interpreter for interpreting applet bytecode, a Java verifier to check for security issues, and run-time systems for executing the applets. In fact, page 896 of the article "Using Netscape 2," copyright 1995, by Mark Brown, which was cited by the Examiner, explains how Netscape runs Java applets. No undue experimentation to implement an applet engine would have been necessary.

On page 11, lines 16-20, the Specification describes a configuration engine that configures the operating system based on configuration data such as TCP protocol data, domain name server addresses, etc. received from the master server. On page 12, lines 1-4, the Specification identifies a configuration engine that configures the web browser based on configuration data such as home page addresses, bookmarks, caching data, user preferences, etc. received from the master server. On page 12, lines 6-9, the Specification identifies a configuration engine that configures the applet engine based on applet engine configuration data received from the master server. On page 12, lines 9-13, the Specification identifies a configuration engine that configures an applet using applet-specific configuration data from the master server. The Specification exemplifies applet-specific configuration data for an e-mail applet as possibly including the user's e-mail address, name, preferred signature block, and user interface parameters.

Installer software for loading and optimizing a software program onto a system existed at the time this application was filed. One skilled in the art at the time this application was filed had been aware that these systems optimize software programs by loading configuration information into appropriate files, which are later retrieved by the programs to set software program parameters. In the case of configuring an applet, one skilled in the art would have recognized that parameter passing techniques were also available, as evidenced by pages 901-902 of the Brown article cited above. One skilled in the art would know "how" to build this configuration engine. No undue experimentation to implement a configuration engine would have been necessary.

On page 13, lines 13-18, the Specification states that an applet host engine that manages and responds to the execution of downloaded applets. The Specification further states that the applet host engine may initiate and execute corresponding servlets to respond to applet requests. Applet-handling server code such as servlets were known before the filing of this application. According to the article "Servlet Sandbox" published on November 11, 1996, servlets are used to extend the web server, similar to the way that applets extend the web browser. As indicated by the attached article "Java Servlet Application Programming Interface White Paper" published on November 12, 1996, servlets are Java objects that extend the functionality of information servers and can be thought of as server-side applets. The article identifies the difference between applets and servlets by indicating that servlets are faceless objects without a user interface. As indicated by the article "Jeeves Alpha 2 Release Notes" published on November 11, 1996, servlets may provide user interface capabilities. As indicated by the attached article "Overview of the Java HTTP Server Architecture" published on November 12, 1996, servlets are similar to applets in that they are object bytecodes that can be dynamically loaded off the net. The article further indicates that servlets are identified by a URL address or class name. According to the article "Frequently Asked Questions (FAQ) about Jeeves" published on October 2, 1996, servlets serve as platform independent dynamically loadable pluggable helper bytecode objects on the server side. As admitted by the Examiner, CGI programs, as an alternative to servlets, were available at the time the application was filed to manage requests made by an applet. No undue experimentation to implement an applet host engine would have been necessary.

In paragraph 2, the Examiner further asserts that it is not clear how the term "engine" differs from other programs or processes. The Specification employs the term "engine" on several occasions. For example, the Specification describes "a communications engine," "an OS configuration engine," "an applet engine," etc. As illustrated in the embodiment of Figure 2, each of these engines are stored in RAM. This clearly indicates that the engines in this embodiment are software code. Further, the Specification states on page 11 that the "communications engine" generates and transfers message packets to and from the Internet, on page 11 that the "OS configuration engine" configures the operating system, and on page 12 that the applet engine handles the execution of the downloaded applets. Still further, the Computer Dictionary, 2nd edition, published by Microsoft Press (1994) defines the term "engine" to include "The portion of a program that determines how the program manages and manipulates data." Still even further, the Applicant states on page 22 that software portions of the invention may use Application Specific Integrated Circuits (ASICs) or equivalents. It is therefore clear from the Specification, the figures and common usage that the term "engine" describes program code, ASICs or equivalents for causing a computer to perform a specific function.

The Examiner still further asserts that the Specification shows servlets, that the claims do not have anything to do with servlets except by implication, that servlets are not fully taught by the Specification, and that CGI programs are equivalent of servlets.

First, Applicant is not required to claim everything shown in the Specification.

Second, the Examiner seems to believe that Applicant is trying to claim applets and servlets. This is not the case. Applicant's invention was implemented using applet and servlet technology. But, Applicant would like to point out that applet and servlet technology was described in the Specification as an example of technology for making and using the invention and readily available at the time the application was filed as evidenced by the attached articles. The Examiner even admits that CGI programs are a known alternative to servlets.

Third, Applicant states in the originally filed Specification that "'applets' correspond to all downloadable and executable or interpretable programs for use in a distributed environment such as in the ActiveX™ distributed environment." Accordingly, it is evident that the Specification intended the term "applets" to include Java applets, ActiveX controls, downloaded stand-alone applications, plugins, etc. To minimize confusion, however, Applicant is amending the Specification and claims to use the more general term "Downloadable" or "Downloadable code." Further, the Specification states "To initiate execution of the service request, the master server 130 may use servlets or agents." It is therefore evident that the Specification intended servlets as an example of code for responding to requests by an applet.

Fourth, per MPEP 2164.01(b) and In re Fisher, 427 F.2d 833, 839, 166 USPQ 18, 24 (CCPA 1970), "As long as the Specification discloses at least one method for making and using the claimed invention that bears a reasonable correlation to the entire scope of the claim, then the enablement requirement of 35 U.S.C. 112 is satisfied." Applicant has described a method of making and using the invention, namely, using downloadable code such as Java applets and corresponding server code such as Java servlets to enable access to a service across the network. Accordingly, for at least the above reasons, Applicant respectfully submits that the written description and enablement requirements of 35 USC § 112, first paragraph, have been satisfied.

35 USC § 112 Rejections to the Claims

In paragraph 3, the Examiner rejected claims 31-38 and 40-42 under 35 USC §112, first paragraph for the reasons set forth in the objection to the Specification. Applicant has deleted claims 31-37. Applicant submits that claims 38 and 40-42 are valid for the same reasons as set forth above.

35 USC § 103 Rejections to the Claims

In paragraph 6, the Examiner rejected claims 1-42 under 35 USC § 103 as unpatentable over Foley in view of Brown. The Examiner asserted that Foley teaches a portfolio management system which allows users to manage, create, edit, debug and compile components or projects such as applets or class libraries. The Examiner then asserted that Foley teaches all the means of claim 22 except the configuration means which configures the attributes of the client. The Examiner further asserted that Brown teaches this configuration means.

Both Foley and Brown are directed to software development tools for web page designers. Foley enables the web page designer to download remote portfolio files from the internet as needed. Brown enables the web page designer to configure an applet during the design phase. However, the present invention is directed at *user-based* systems and methods for enabling an end user *to access services on a network* from any connected terminal that executes a web browser having a Downloadable execution engine. The present invention is also directed at *user-based* systems and methods for personalizing the remote client to include user-preferred configuration parameters. Applicant directs the Examiner's attention to each of the independent claims, namely, claims 1, 8, 15, 22, 29, 30, 38, 43 and 44. Neither Brown nor Foley teaches the limitations "establishing a communications link between the client and a service using the Downloadable code" and "interfacing with the selected service using the Downloadable code" recited in claim 1 and similarly recited in the remaining independent claims.

For at least these reasons, Applicant submits that claims 1-30 and 38-50 are non-obvious over the art of record, and respectfully requests the rejection under 35 USC § 103 be withdrawn.

If the Examiner has any questions or needs any additional information, the Examiner is invited to telephone the undersigned attorney at (650) 843-3392.

If for any reason an insufficient fee has been paid, the Assistant Commissioner is hereby authorized to charge the insufficiency to Deposit Account No. 05-0150.

Respectfully Submitted,
Riggins et al.

Dated: August 12, 1998

Graham & James LLP
600 Hansen Way
Palo Alto, CA 94304-1043
Telephone: (650) 856-6500
Facsimile: (650) 856-3619

Marc A. Sockol
Attorney for Applicants
Reg. No. 40,823